

Article

PSO with Dynamic Adaptation of Parameters for Optimization in Neural Networks with Interval Type-2 Fuzzy Numbers Weights

Fernando Gaxiola ¹, Patricia Melin ², Fevrier Valdez ^{2,*}, Juan R. Castro ³ and Alain Manzo-Martínez ¹

¹ Engineering Faculty, University Autonomous of Chihuahua, Chihuahua 31125, Mexico; lgaxiola@uach.mx (F.G.); amanzo@uach.mx (A.M.-M.)

² Division of Graduate Studies, Tijuana Institute of Technology, Tijuana 22414, Mexico; pmelin@tectijuana.mx

³ Chemical Faculty of Sciences and Engineering, University Autonomous of Baja California, Tijuana 14418, Mexico; jrcastror@uabc.edu.mx

* Correspondence: fevrier@tectijuana.mx

Received: 4 December 2018; Accepted: 9 January 2019; Published: 24 January 2019



Abstract: A dynamic adjustment of parameters for the particle swarm optimization (PSO) utilizing an interval type-2 fuzzy inference system is proposed in this work. A fuzzy neural network with interval type-2 fuzzy number weights using S-norm and T-norm is optimized with the proposed method. A dynamic adjustment of the PSO allows the algorithm to behave better in the search for optimal results because the dynamic adjustment provides good synchrony between the exploration and exploitation of the algorithm. Results of experiments and a comparison between traditional neural networks and the fuzzy neural networks with interval type-2 fuzzy numbers weights using T-norms and S-norms are given to prove the performance of the proposed approach. For testing the performance of the proposed approach, some cases of time series prediction are applied, including the stock exchanges of Germany, Mexican, Dow-Jones, London, Nasdaq, Shanghai, and Taiwan.

Keywords: particle swarm optimization; fuzzy numbers; type-2 fuzzy weights; neural networks; backpropagation; time series prediction

1. Introduction

In time series prediction problems, the main objective is to obtain results that approximate the real data as closely as possible with minimum error. The use of intelligent systems for working with time series problems is widely utilized [1,2]. For example, Zhou et al. [3] used a dendritic mechanism in a neural model and phase space reconstruction (PSR) for the prediction of a time series, and Hrasko et al. [4] presented a prediction of a time series with a Gaussian-Bernoulli restricted Boltzmann machine hybridized with the backpropagation algorithm.

For optimization problems, the principle objective is to work in a search space to encounter an optimal choice among a set of potential solutions. In many cases, the search space is too wide, which means that the time used for obtaining an optimal solution is very extensive. To improve the optimization and search problems, a set of methods of computational intelligence was the focus of some recent work in improving the solving of optimization and search problems [5–7]. These methods can achieve competitive results, but not the best solutions. Furthermore, heuristic algorithms can be used like alternative methods. Although these algorithms are not guaranteed to find the best solution, they are able to obtain an optimal solution in a reasonable time.

The main contribution is the dynamic adaptation of the parameters for particle swarm optimization (PSO) used to optimize parameters for a neural network with interval type-2 fuzzy

numbers weights with different T-norms and S-norms, proposed in Gaxiola et al. [8] and used to perform the prediction of financial time series [9–11].

The adaptation of parameters is performed with a Type-2 fuzzy inference system (T2-FIS) to adjust the c_1 and c_2 parameters for the particle swarm optimization. This adaptation is based on the work of Olivas et al. [12], who used interval type-2 fuzzy logic to improve the convergence and diversity of PSO to optimize the minimum for several mathematical functions. In this work, control of the c_1 and c_2 parameters is performed using fuzzy rules to achieve a good dynamic adaptation [13,14]. The use of type-2 fuzzy logic gives a better performance in our work and this is justified by recent research from Wu and Tan [15] and Sepulveda et al. [16], who demonstrated that interval type-2 fuzzy systems perform better than type-1 fuzzy systems.

The neural network with interval type-2 fuzzy numbers weights optimized with the proposed approach is different from others works using the adjustment of weights in neural networks [17,18], such as James and Donald [19,20].

A comparison of the performance of the traditional neural network compared to the proposed optimization for the fuzzy neural network with interval type-2 fuzzy numbers weights is performed in this paper. In this case, a prediction for a financial time series is used to verify the efficiency of the proposed method.

The use of T-norms and S-norms of sum-product can be seen in Hamacher [21,22]. Olivas [23] developed a new method for PSO of parameter adaptation using fuzzy logic. In their work, the authors improved the performance of the traditional PSO. We decided to use the parameter adaptation because in previous works, we have had good results with other metaheuristic methods. For instance, in [24], we proposed a gravitational search algorithm (GSA) with parameter adaptation to improve the original GSA metaheuristic algorithm. The next section states the theoretical approach to the work in the paper. Section 3 presents the background research performed to adjust the parameters in PSO, as well as research performed in the neural network area with fuzzy numbers. Section 4 explains the proposed approach and a description of the problem to be solved in the paper. Section 5 presents the simulation results for the proposed approach. Section 6 presents a discussion of the results of the experiments. Finally, in Section 7, conclusions are shown.

2. Theoretical Basement

2.1. Particle Swarm Optimization (PSO)

Kennedy and Eberhart [25] introduced a bio-inspired algorithm based on the behavior of a swarm of particles “flying” (moving around) through a multidimensional space search, and the potential solutions to the problem are represented as particles [26]. The dynamics of the algorithm are performed by adjusting the position of each particle based on the experience acquired for each particle and its neighbors [27,28]. The search in PSO is defined by the equations to update the position (Equation (1)) and velocity (Equation (2)) of each particle, respectively:

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (1)$$

where:

x_i is the particle i .

t is the current iteration.

v_i is the velocity of the particle i .

$$v_i(t+1) = v_{ij}(t) + c_1 r_1(t) [y_{ij}(t) - x_{ij}(t)] + c_2 r_2(t) [\hat{y}_j(t) - x_{ij}(t)] \quad (2)$$

where:

v_i is the velocity of the particle i in the dimension j .

c_1 is the cognitive factor (importance of the best previous position of the particle).

c_2 is the social factor (importance of the best global position of the swarm).

r_1, r_2 are random values in the range of $[0, 1]$.
 y_{ij} is the best position of the particle i in the dimension j .
 x_{ij} is the current position of the particle i in the dimension j .
 \hat{y}_j is the best global position of the swarm in the dimension j .

In Figure 1, the movements of a particle in the search space according to Equations (1) and (2) are presented.

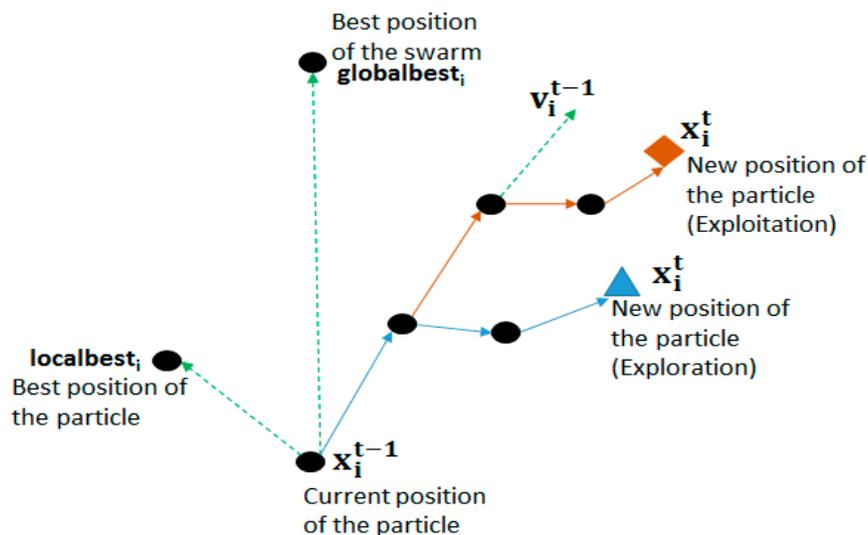


Figure 1. Movement of the particle in the search space.

The triangle and rhombus points represent the movement of the particle with a change in the parameters, c_1 and c_2 . The triangle point is the result for the case when $c_1 > c_2$, in this case, the swarm performs exploration and “fly” in the search space to find a best area which allows the obtainment of the global best position; so, the exploration allows the particles to perform long movements for travelling in all the search space. The rhombus point is the result for the case when $c_1 < c_2$, in this case, the swarm uses exploitation and “fly” in the best area of the search space with short movements to perform an exhaustive search in the best area.

2.2. Type-2 Fuzzy Systems

Zadeh [29] introduced the approach of type-2 fuzzy sets as an extension of the type-1 fuzzy sets. Later, Mendel and John [30] defined the type-2 fuzzy set as follows:

$$A^\sim = \{[(x, u), \mu_{A^\sim}(x, u)] | \forall x \in X, \forall u \in J_x \subseteq [0, 1]\} \tag{3}$$

where
 A^\sim is a type-2 fuzzy set.
 $\mu_{A^\sim}(x, u)$ is a type-2 membership function.
 J_x is called primary membership function of A^\sim .

The type-2 fuzzy set (T2FS) consists of generating a bounded region of uncertainty in the primary membership, called the footprint of uncertainty (FOU) [31]. The interval type-2 fuzzy sets are a particular case of T2FS in which the FOU is generated by the union of two type-1 membership functions (MF), a lower MF (LMF) and an upper MF (UMF) [32]. For IT2FS, the defuzzification is performed by the calculation of the centroid of the FOU founded in Karnik and Mendel [33]. The use of a T2FS allows the handling of more uncertainty and provides fuzzy systems with more robustness than a type-1 fuzzy sets [34–36].

2.3. Fuzzy Neural Network

The area of fuzzy neural networks consists in developing hybrid algorithms that implement the approach of neural networks architectures with fuzzy logic theory [17,18,37,38].

3. Antecedents Development

The PSO bio-inspired algorithm has received many improvements in their performance by many researchers. The most important modifications made to PSO are to increment the diversity of the swarm and to enhance the convergence [39,40].

Olivas et al. [12] proposed a method to dynamically adjust the parameters of particle swarm optimization using type-2 and type-1 fuzzy logic.

Muthukaruppan and Er [41] presented a hybrid algorithm of particle swarm optimization using a fuzzy expert system in the diagnosis of coronary artery disease.

Taher et al. [42] developed a fuzzy system to perform an adjustment of the parameters, c_1 , c_2 , and w (inertia weight), for the particle swarm optimization.

Wang et al. [43] presented a PSO variant utilizing a fuzzy system to implement changes in the velocity of the particle according to the distance between all particles. If the distance is small, the velocity of some particles is modified drastically.

Hongbo and Abraham [44] proposed a new parameter, called the minimum velocity threshold, to the equation used to calculate the velocity of the particle. The approach consists in that the new parameter performs the control in the velocity of the particles and applies fuzzy logic to make the new parameter dynamically adaptive.

Shi and Eberhart [45] presented a dynamic adaptive PSO to adjust the inertia weight utilizing a fuzzy system. The fuzzy system works with two input variables, the current inertia weight and the current best performance evaluation, and the change of the inertia weight is the output variable.

In neural networks, the use of fuzzy logic theory to obtain hybrid algorithms has improved the results for several problems [46–48]. In this area, some significant papers of works with fuzzy numbers are presented [49,50]:

Dunyak et al. [51] proposed a fuzzy neural network that works with fuzzy numbers for the obtainment of new weights (inputs and outputs) in the step of training. Coroianu et al. [52] presented the use of the inverse F-transform to obtain the optimal fuzzy numbers, preserving the support and the convergence of the core.

Li Z. et al. [53] proposed a fuzzy neural network that works with two fuzzy numbers to calculate the results with operations, like subtraction, addition, division, and multiplication. Fard et al. [54] presented a fuzzy neural network utilizing sum and product operations for two interval type-2 triangular fuzzy numbers in combination with the Stone-Weierstrass theorem.

Molinari [55] proposed a comparison of generalized triangular fuzzy numbers with other fuzzy numbers. Asady [56] presented a comparison of a new method for approximation trapezoidal fuzzy numbers with methods of approximation in the literature.

Figueroa-García et al. [57] presented a comparison for different interval type-2 fuzzy numbers performing distance measurement. Requena et al. [58] proposed a fuzzy neural network working with trapezoidal fuzzy numbers to obtain the distance between the numbers and utilized a proposed decision personal index (DPI).

Valdez et al. [59] proposed a novel approach applied to PSO and ACO algorithms. They used fuzzy systems to dynamically update the parameters for the two algorithms. In addition, as another important work, we can mention Fang et al. in [60], who proposed a hybridized model of a phase space reconstruction algorithm (PSR) with the bi-square kernel (BSK) regression model for short term load forecasting. Also, it is worth mentioning the interesting work of Dong et al. [61], who proposed a hybridization method using the support vector regression (SVR) model with a cuckoo search (CS) algorithm for short term electric load forecasting.

4. Proposed Method and Problem Description

In this work, the optimization for interval type-2 fuzzy number weights neural networks (IT2FNWNN) using particle swarm optimization (PSO) is proposed. The PSO is dynamically adapted utilizing interval type-2 fuzzy systems. A comparison of the traditional neural network and the IT2FNWNN optimized with PSO is performed. The operations in the neurons are obtained with calculation of T-norms and S-norms of the sum-product, Hamacher and Frank [62–64].

For testing the proposed method, the prediction of the financial time series of the stock exchange markets of Germany, Mexican, Dow-Jones, London, Nasdaq, Shanghai, and Taiwan is performed. The databases consisted of 1250 data for each time series (2011 to 2015).

The architecture of the traditional neural network used in this paper (see Figure 2) consists of 16 neurons for the hidden layer and one neuron for the output layer.

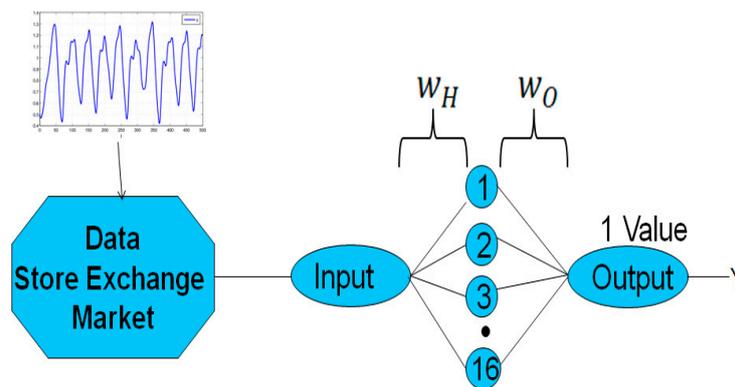


Figure 2. Scheme of the architecture of the traditional neural network.

The architecture of the interval type-2 fuzzy number weight neural network (IT2FNWNN) has the same structure of the traditional neural network, with 16, 39, and 19 neurons in the hidden layer for the sum-product, Hamacher and Frank, respectively, and one neuron for the output layer (see Figure 3) [8].

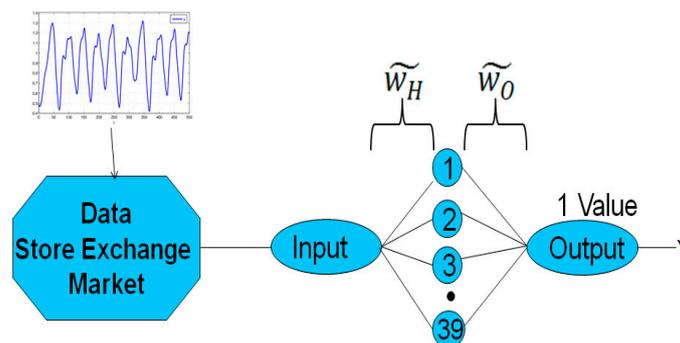


Figure 3. Scheme of the architecture of the interval type-2 fuzzy number weight neural network.

The IT2FNWNN work with fuzzy numbers weights in the connections between the layers, and these weights are represented as follows [8]:

$$\tilde{w} = [w, \bar{w}] \tag{4}$$

where \underline{w} and \bar{w} are obtained using the Nguyen-Widrow algorithm [65] in the initial execution of the network.

The linear function as the activation function for the output neuron and the secant hyperbolic for the hidden neurons are applied. The S-Norm and T-Norm for obtaining the lower and upper outputs

of the neurons are applied, respectively. For the lower output, the S-Norm is used and for the upper output, the T-Norm is utilized. The T-Norms and S-Norms used in this work are described as follows:

Sum-product:

$$TNorm(\overline{Net}, \overline{Net}) = \overline{Net} \times \overline{Net} \tag{5}$$

$$SNorm(\overline{Net}, \overline{Net}) = \overline{Net} + \overline{Net} - TNorm(\overline{Net}, \overline{Net}) \tag{6}$$

Hamacher: for $\gamma > 0$.

$$TNormH(\overline{Net}, \overline{Net}, \gamma) = \frac{\overline{Net} \times \overline{Net}}{\gamma + (1 - \gamma)(\overline{Net} + \overline{Net} - \overline{Net} \times \overline{Net})} \tag{7}$$

$$SNormH(\overline{Net}, \overline{Net}, \gamma) = \frac{\overline{Net} + \overline{Net} + (\gamma - 2)(\overline{Net} \times \overline{Net})}{1 + (\gamma - 1)(\overline{Net} \times \overline{Net})} \tag{8}$$

Frank: for $s > 0$.

$$TNormF(\overline{Net}, \overline{Net}, s) = \log_s \left[\frac{1 + (s^{\overline{Net}} - 1)(s^{\overline{Net}} - 1)}{s - 1} \right] \tag{9}$$

$$SNormF(\overline{Net}, \overline{Net}, s) = 1 - \log_s \left[\frac{1 + (s^{1-\overline{Net}} - 1)(s^{1-\overline{Net}} - 1)}{s - 1} \right] \tag{10}$$

The back-propagation algorithm using gradient descent and an adaptive learning rate is implemented for the neural networks.

A modification of the back-propagation algorithm that allows operations with interval type-2 fuzzy numbers to obtain new weights for the forthcoming epochs of the fuzzy neural network is proposed [8].

The particle swarm optimization (PSO) is used to optimize the number of neurons for the IT2FNWNN for operations with S-norm and T-norm of Hamacher and Frank. Also, for Hamacher, the parameter, γ , is optimized, and for Frank, the parameter, s , is optimized.

The parameters used in the PSO to optimize the IT2FNWNN are presented in Table 1. The same parameters and design are applied to optimize all the IT2FNWNN for all the financial time series.

Table 1. Parameters of PSO to optimize IT2FNWNN.

Parameters	Values
Particles	50
Dimensions	2
Iterations	50
Inertia Weight	0.1
Constriction Coefficient (C)	1
R1, R2	Random in [0, 1]
c_1, c_2	Dynamic adjust IT2FIS

In the literature [27], the recommended values for the c_1 and c_2 parameters are in the interval of 0.5 and 2.5; also, the change of these parameters in the iterations of the PSO can generate better results.

In the PSO used to optimize the IT2FNWNN, the parameters, c_1 and c_2 , from Equation (2) are selected to perform the dynamic adjustment utilizing an interval type-2 fuzzy inference system (IT2FIS). This selection is in the base as these parameters allow the particles to generate movements for performing the exploration or exploitation in the search space. The structure of the IT2FIS utilized in the PSO is presented in Figure 4.

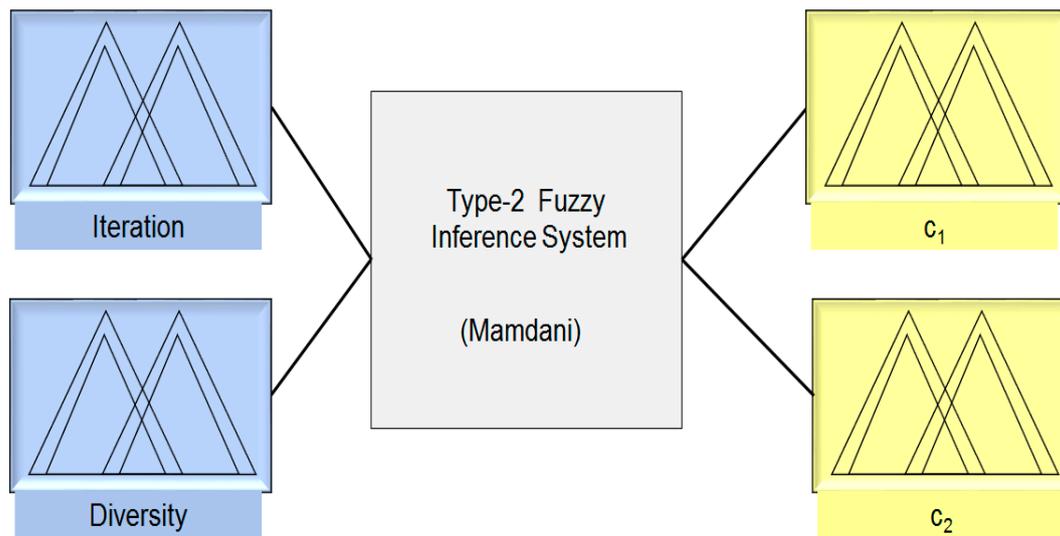


Figure 4. Structure of type-2 fuzzy inference system for the PSO.

Based on the literature [12,66,67], the inputs selected for the IT2FIS are the diversity of the swarm and the percentage of iterations. Also, in [68], a problem of online adaptation of radial basis function (RBF) neural networks is presented. The authors present a new adaptive training, which is able to modify both the structure of the network (the number of nodes in the hidden layer) and the output weights.

The diversity input is defined by Equation (11), in which the degree of the dispersion of the swarm is calculated. This means that for less diversity, the particles are closer together, and for high diversity, the particles are more separated. The diversity equation can be taken as the average of the Euclidean distances amongst each particle and the best particle:

$$Diversity(t) = \frac{1}{n_s} \sum_{i=1}^{n_s} \sqrt{\sum_{j=1}^{n_x} [x_{ij}(t) - \bar{x}_j(t)]^2} \tag{11}$$

where:

$x_{ij}(t)$: represents the particle in the iteration t .

$\bar{x}_j(t)$: represents the best particle in the iteration t .

For the diversity variable, normalization is performed based on Equation (12), taking values in the range of 0 to 1:

$$DiversityNorm = \begin{cases} 0, & \text{if } minDiver = maxDiver \\ DiverNorm, & \text{if } minDiver \neq maxDiver \end{cases} \tag{12}$$

where:

$minDiver$: Minimum Euclidian distance for the particle.

$maxDiver$: Maximum Euclidian distance for the particle.

$DiverNorm$: Value obtained with Equation (13).

$$DiverNorm = \frac{Diversity - minDiver}{maxDiver - minDiver} \tag{13}$$

The membership functions for the input diversity are presented in Figure 5.

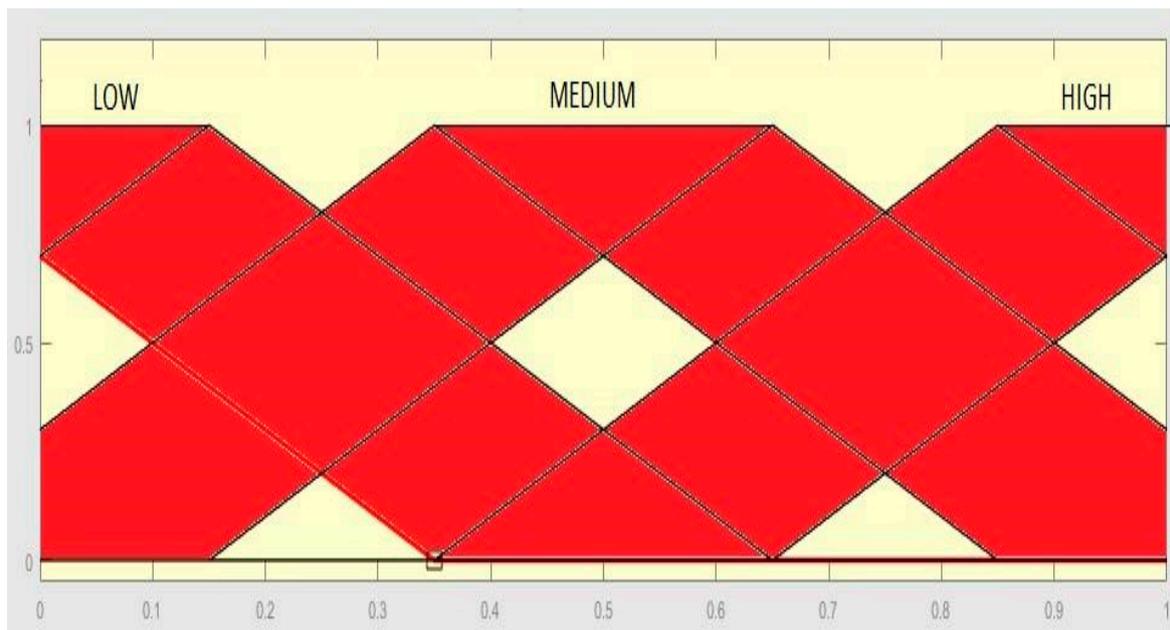


Figure 5. Membership function for the input diversity.

The iteration input is considered as a percentage of iterations during the execution of the PSO, the values for this input are in the range from 0 to 1. In the start of the algorithm, the iteration is considered as 0% and is increased until it reaches 100% at the end of the execution. The values are obtained as follows:

$$Iteration = \frac{Currentiteration}{Totalofiterations} \tag{14}$$

The membership functions for the input iteration are presented in Figure 6.

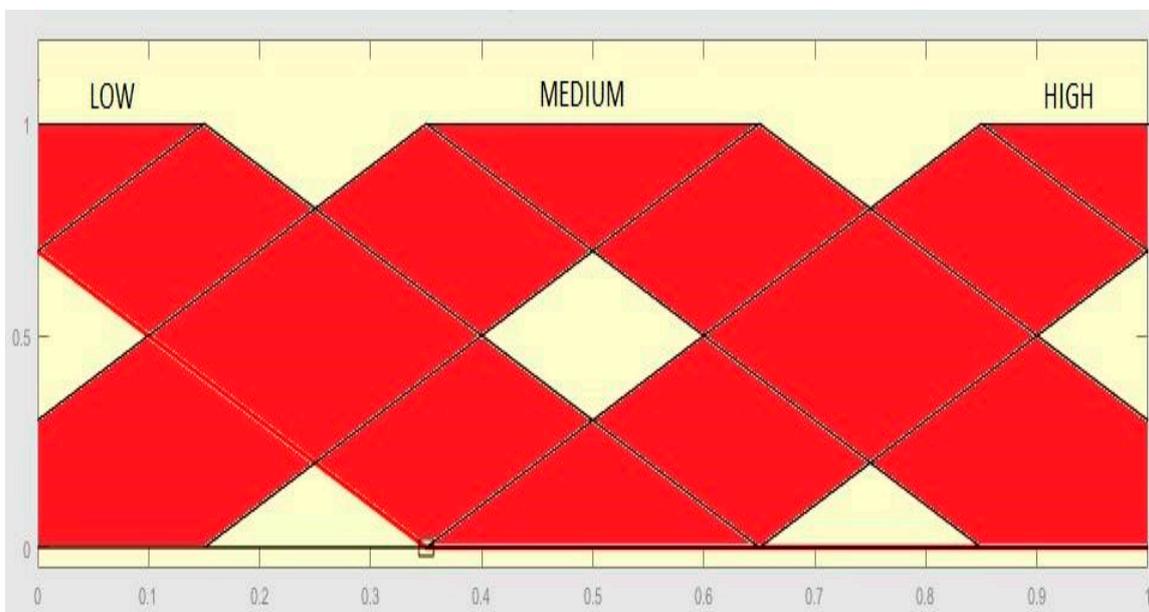


Figure 6. Membership function for the input iteration.

The membership functions for the outputs, c_1 and c_2 , are presented in Figures 7 and 8, respectively. The values for the two outputs are in the range from 0 to 3.

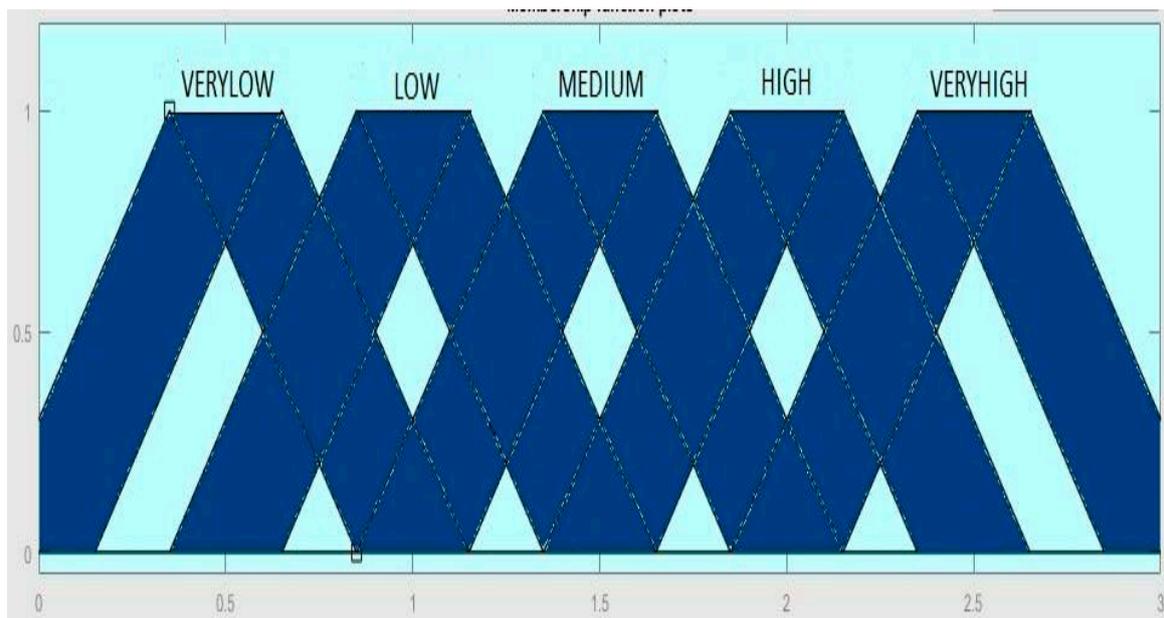


Figure 7. Membership function for the output, c_1 .

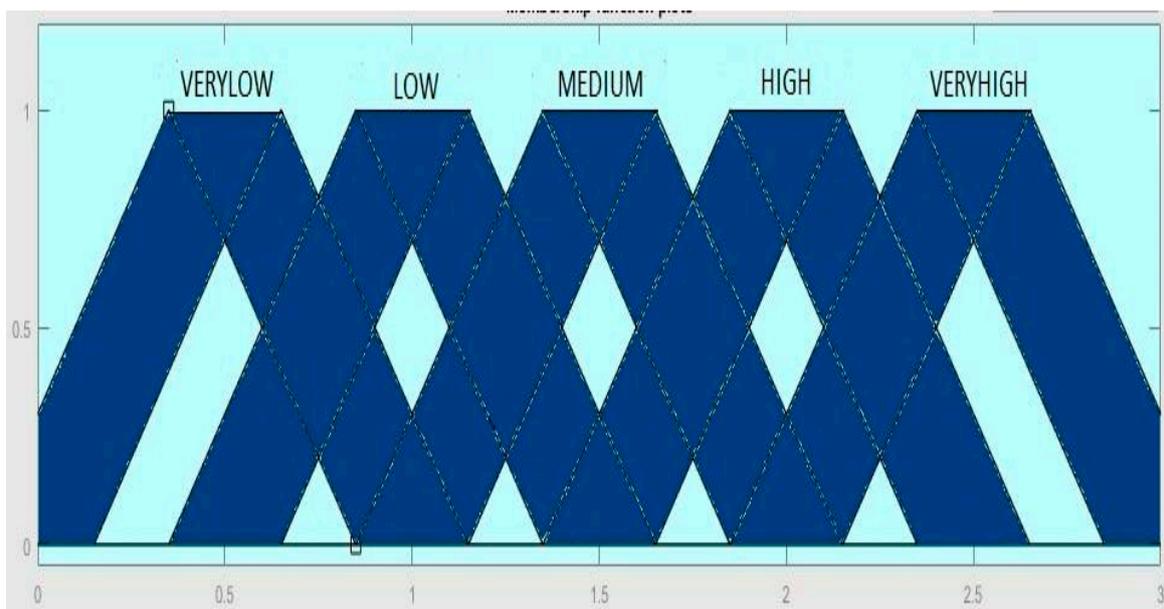


Figure 8. Membership function for the output, c_2 .

Based on the literature of the behavior for the c_1 and c_2 parameters described in Section 2, for the dynamic adjustment of the c_1 and c_2 parameters in each iteration of the PSO, a fuzzy rules set with nine rules is designed as follows:

1. If (Iteration is Low) and (Diversity is Low) then (c_1 is VeryHigh) (c_2 is VeryLow).
2. If (Iteration is Low) and (Diversity is Medium) then (c_1 is High) (c_2 is Medium).
3. If (Iteration is Low) and (Diversity is High) then (c_1 is High) (c_2 is Low).
4. If (Iteration is Medium) and (Diversity is Low) then (c_1 is High) (c_2 is Low).
5. If (Iteration is Medium) and (Diversity is Medium) then (c_1 is Medium) (c_2 is Medium).
6. If (Iteration is Medium) and (Diversity is High) then (c_1 is Low) (c_2 is High).
7. If (Iteration is High) and (Diversity is Low) then (c_1 is Medium) (c_2 is VeryHigh).
8. If (Iteration is High) and (Diversity is Medium) then (c_1 is Low) (c_2 is High).

9. If (Iteration is High) and (Diversity is High) then (c_1 is VeryLow) (c_2 is VeryHigh).

In Figure 9, a block diagram of the procedure for the proposed approach is presented.

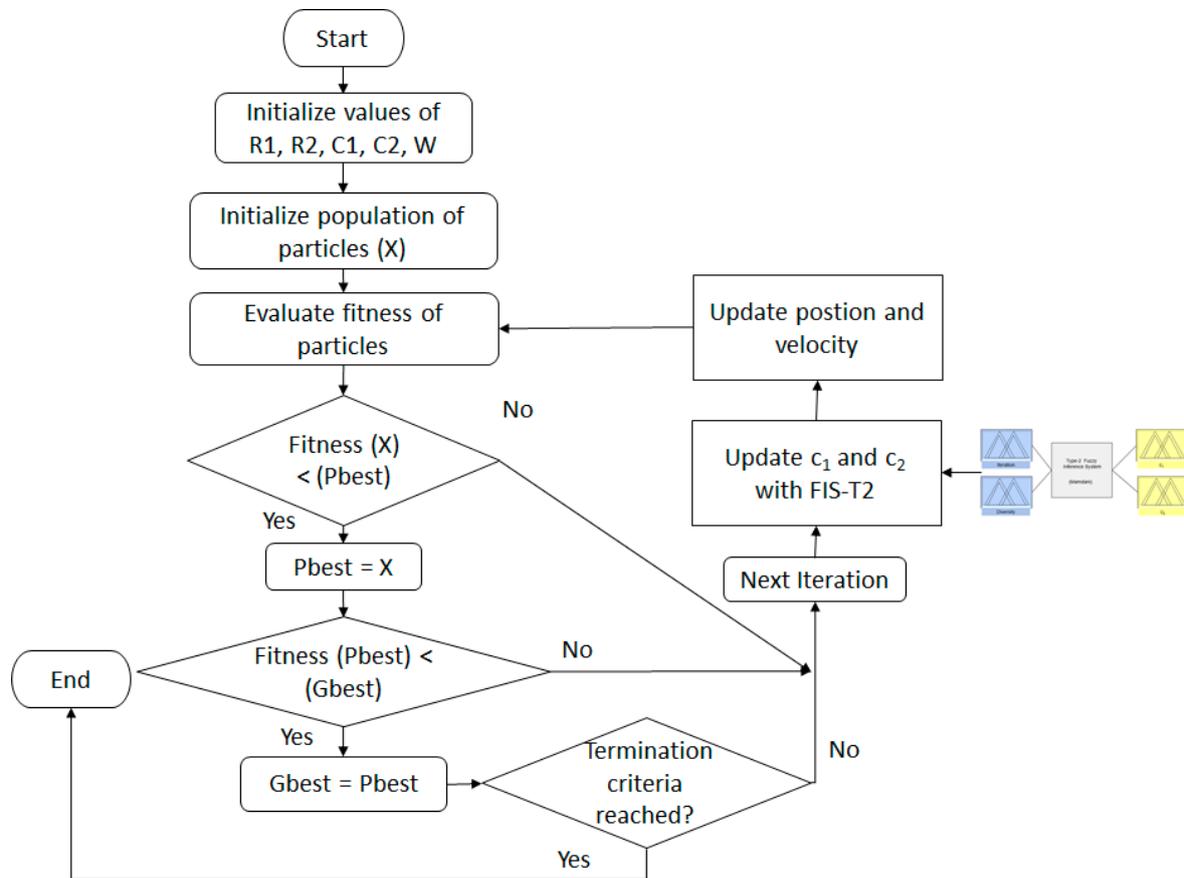


Figure 9. Block diagram of the proposed approach.

5. Simulation Results

We achieved the experiments for the financial time series of the stock exchange markets of Germany, Mexico, Dow-Jones, London, Nasdaq, Shanghai, and Taiwan, and for all experiments, we used 1250 data points. In this case, 750 data points are considered for the training stage and 500 data points for the testing stage.

In Table 2, the results of the experiments for the financial time series of stock exchange markets for the traditional neural network with 16 neurons in the hidden layer are presented.

Table 2. Results of the traditional neural network for the financial series.

Error	Results	Germany	Mexican	Dow-Jones	London	Nasdaq	Shanghai	Taiwan
MAE	Best	1089.19	733.50	920.89	167.05	468.81	361.95	383.00
	Average	1138.48	755.34	956.03	168.76	470.43	367.90	412.52
RMSE	Best	1264.68	917.93	1075.70	200.13	550.95	561.29	458.28
	Average	1332.28	935.58	1109.73	203.29	558.95	580.00	480.11

The results of the experiments are obtained with the mean absolute error (MAE) and root mean standard error (RMSE). Thirty experiments with the same parameters and conditions to obtain the average error were performed, but only the best result and average of the 30 experiments are presented.

In Table 3, the results of the experiments for the financial time series of the stock exchange for the IT2FNWNN with 16 neurons in the hidden layer for the S-norm and the T-norm of the sum of the product are presented (IT2FNWNNSp).

Table 3. Results of the IT2FNWNN with the S-norm and T-norm of the sum of the product for the financial time series.

Error	Results	Germany	Mexican	Dow-Jones	London	Nasdaq	Shanghai	Taiwan
MAE	Best	1038.84	739.26	736.26	155.51	585.70	391.21	321.86
	Average	1148.27	760.49	937.09	159.20	725.07	403.97	339.18
RMSE	Best	1338.21	915.45	840.21	192.14	786.84	458.75	499.62
	Average	1414.93	939.70	1098.61	196.65	884.07	474.99	615.84

In Table 4, the results of the experiments for the financial time series of the stock exchange markets for the IT2FNWNN with 39 neurons in the hidden layer for the S-norm and T-norm of Hamacher are presented (IT2FNWNNH). The value for the parameter, γ , in the Hamacher IT2FNWNN without optimization is 1.

Table 4. Results of the IT2FNWNN with the S-norm and T-norm of Hamacher for the financial time series.

Error	Results	Germany	Mexican	Dow-Jones	London	Nasdaq	Shanghai	Taiwan
MAE	Best	1055.06	730.13	686.05	151.76	463.77	314.71	349.78
	Average	1093.54	756.95	787.61	158.96	711.63	334.81	392.70
RMSE	Best	1216.24	912.65	800.68	188.46	570.36	569.32	449.41
	Average	1334.42	945.19	925.43	198.98	854.65	626.49	472.24

In Table 5, the results of the experiments for the financial time series of the stock exchange for the IT2FNWNN with 19 neurons in the hidden layer for the S-norm and T-norm of Frank are presented (IT2FNWNNF). The value for the parameter, s , in the Frank IT2FNWNN without optimization is 2.8.

Table 5. Results of the IT2FNWNN with the S-norm and T-norm of Frank for the financial time series.

Error	Results	Germany	Mexican	Dow-Jones	London	Nasdaq	Shanghai	Taiwan
MAE	Best	1058.87	730.96	853.69	153.35	663.02	354.07	362.56
	Average	4439.15	756.19	944.12	159.58	741.04	336.64	399.85
RMSE	Best	1256.37	915.84	972.34	186.68	730.97	607.68	448.20
	Average	1402.82	936.95	1101.39	197.71	902.34	640.26	470.09

We performed the optimization of the Hamacher IT2FNWNN and Frank IT2FNWNN for the number of neurons in the hidden layer and the parameters, γ and s , respectively, with PSO using the dynamic adjustment for the c_1 and c_2 parameters of Equation (2), applying the interval type-2 fuzzy inference system of Figure 4 and the parameters of Table 1.

In Table 6, the optimization for each financial time series for the Hamacher IT2FNWNN is presented. We presented the best result of the five experiments performed.

Table 6. Results of the optimization with Dynamic PSO for Hamacher IT2FNWNN for the financial time series.

Financial Time Series	Prediction Error	No. Neurons	Parameter γ
Germany	417.33	39	1.6471
Mexican	744.34	21	0.9997
Dow-Jones	517.58	15	1.9270
London	155.72	22	1.0016
Nasdaq	227.53	56	0.6822
Shanghai	211.40	99	1.7397
Taiwan	236.15	28	1.1997

In Table 7, the results of the experiments for the financial time series of the stock exchange for the IT2FNWNN for the Hamacher S-norm and T-norm optimized with dynamic PSO are presented (IT2FNWNNH-PSO). The value for the number of neurons in the hidden layer and the parameter, γ , in the Hamacher IT2FNWNN are taken from Table 6.

Table 7. Results of the IT2FNWNN with the S-norm and T-norm of Hamacher optimized with dynamic PSO for the financial time series.

Error	Results	Germany	Mexican	Dow-Jones	London	Nasdaq	Shanghai	Taiwan
MAE	Best	417.33	728.61	517.58	154.87	227.53	211.40	236.15
	Average	813.21	751.48	881.97	160.47	342.14	313.86	401.49
RMSE	Best	507.43	909.39	769.15	192.81	283.92	443.85	331.32
	Average	1172.56	930.47	1290.34	197.99	530.34	543.39	472.44

In Table 8, the optimization for each financial time series for the Frank IT2FNWNN is presented. We are only showing the best result of five performed experiments.

Table 8. Results of the optimization with dynamic PSO for Frank IT2FNWNN for the financial time series.

Financial Time Series	Prediction Error	No. Neurons	Parameter γ
Germany	935.77	89	1.7796
Mexican	726.84	22	1.4607
Dow-Jones	768.81	119	3.1526
London	153.60	32	1.2669
Nasdaq	302.19	73	1.2910
Shanghai	277.72	79	1.9962
Taiwan	359.21	124	1.6347

In Table 9, the results of the experiments for the financial time series of the stock exchange for the IT2FNWNN for Frank S-norm and T-norm optimized with dynamic PSO are presented (IT2FNWNNF-PSO). The value for the number of neurons in the hidden layer and the parameter, s , in the Frank IT2FNWNN are taken for Table 8.

Table 9. Results of the IT2FNWNN with the S-norm and T-norm of Frank optimized with dynamic PSO for the financial time series.

Error	Results	Germany	Mexican	Dow-Jones	London	Nasdaq	Shanghai	Taiwan
MAE	Best	935.77	726.84	768.81	153.60	288.70	277.72	359.21
	Average	1050.87	773.60	863.79	163.12	391.39	373.38	399.28
RMSE	Best	1226.07	911.63	979.26	188.63	349.02	491.98	439.93
	Average	1302.99	963.74	1082.81	201.38	469.52	597.62	481.62

In Table 10, a comparison of the best results for the prediction for all financial time series of the stock exchange markets with the traditional neural network (TNN), the interval type-2 fuzzy numbers weights neural network with the sum of the product (IT2FNWNNSp), Hamacher (IT2FNWNNH) and Frank (IT2FNWNNF) without optimization, and Hamacher (IT2FNWNNH-PSO) and Frank (IT2FNWNNF-PSO) optimized with dynamic PSO is presented.

Table 10. Comparison of the best results of financial time series prediction with all neural networks.

Financial Time Series	Error	TNN	IT2FNWNNSp	IT2FNWNNH	IT2FNWNNF	IT2FNWNNH-PSO	IT2FNWNNF-PSO
Germany	MAE	1089.19	1038.84	1055.06	1058.87	417.33	935.77
	RMSE	1264.68	1338.21	1216.24	1256.37	507.43	1226.07
Mexican	MAE	733.50	739.26	730.13	730.96	728.61	726.84
	RMSE	917.93	915.45	912.65	915.84	909.39	911.63
Dow-Jones	MAE	920.89	736.26	686.05	853.69	517.58	768.81
	RMSE	1075.70	840.21	800.68	972.34	769.15	979.26
London	MAE	167.05	155.51	151.76	153.35	154.87	153.60
	RMSE	200.13	192.14	188.46	186.68	192.81	188.63
Nasdaq	MAE	468.81	585.70	463.77	663.02	227.53	288.70
	RMSE	550.95	786.84	570.36	730.97	283.92	349.02
Shanghai	MAE	361.95	391.21	314.71	354.07	211.40	277.72
	RMSE	561.29	458.75	569.32	607.68	443.85	491.98
Taiwan	MAE	383.00	321.86	349.78	362.56	236.15	359.21
	RMSE	458.28	499.62	449.41	448.20	331.32	439.93

In Figures 10–16, the plots of the real data of each stock exchange time series against the predicted data of the best results for the Interval type-2 fuzzy number weights neural network taken from the comparison in Table 10 are presented.

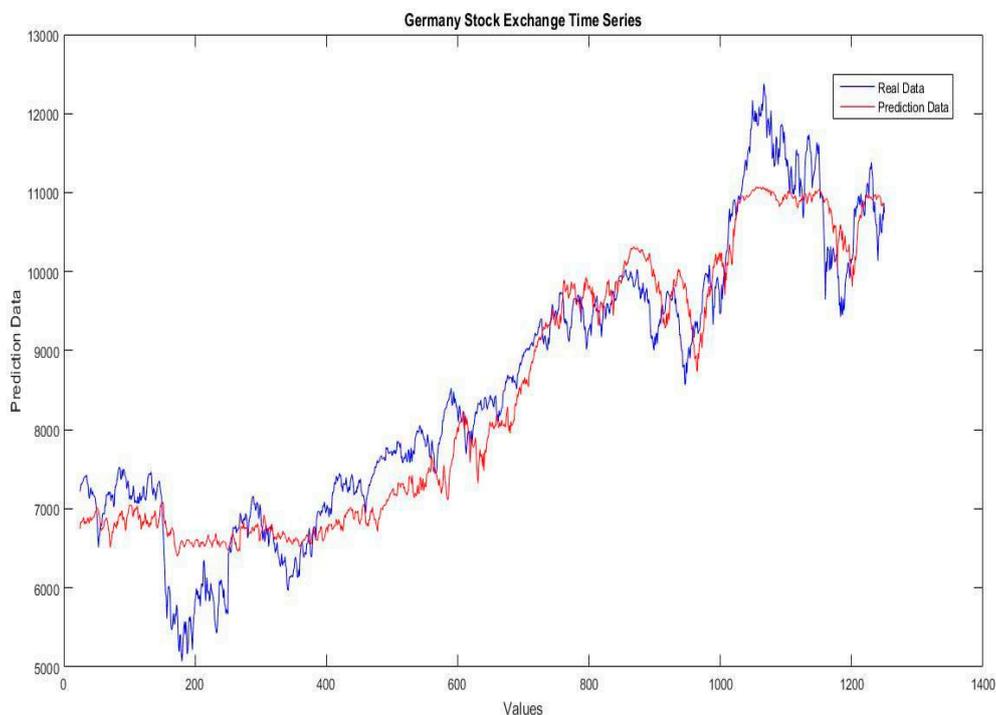


Figure 10. Illustration of the real data against the prediction data of the Germany stock exchange time series for the fuzzy neural network using Hamacher optimized with dynamic PSO.

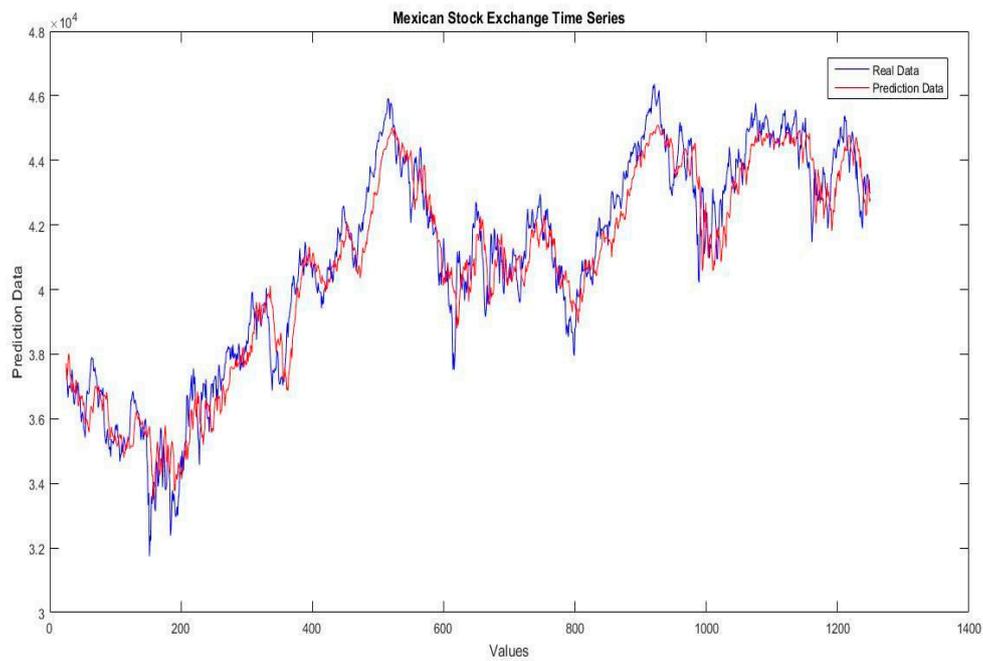


Figure 11. Illustration of the real data against the prediction data of the Mexican stock exchange time series for the fuzzy neural network using Frank optimized with dynamic PSO.

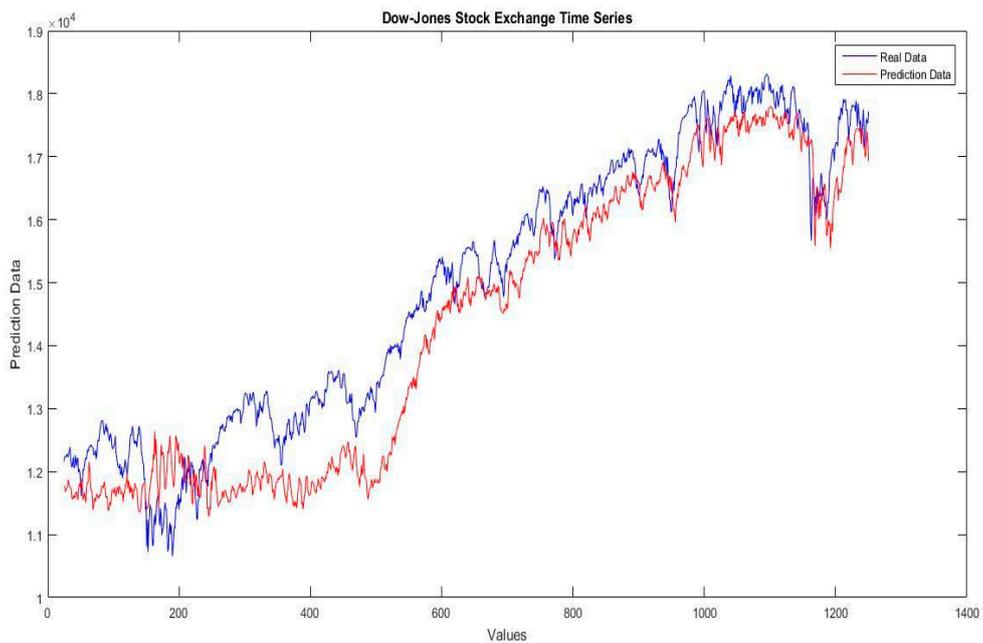


Figure 12. Illustration of the real data against the prediction data of the Dow-Jones stock exchange time series for the fuzzy neural network using Hamacher optimized with dynamic PSO.

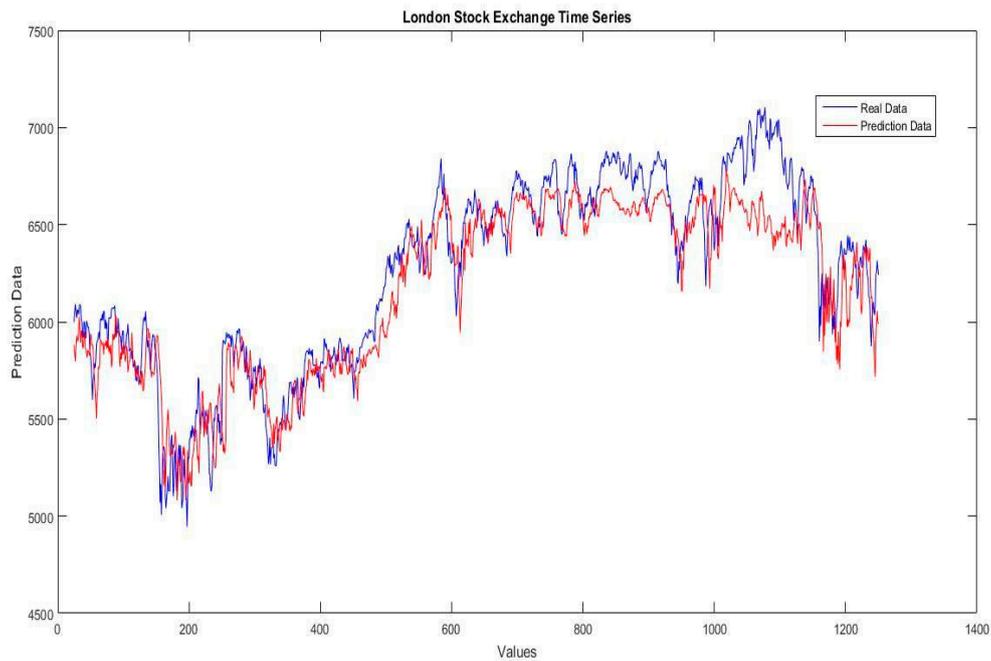


Figure 13. Illustration of the real data against the prediction data of the London stock exchange time series for the fuzzy neural network using Hamacher without optimization.

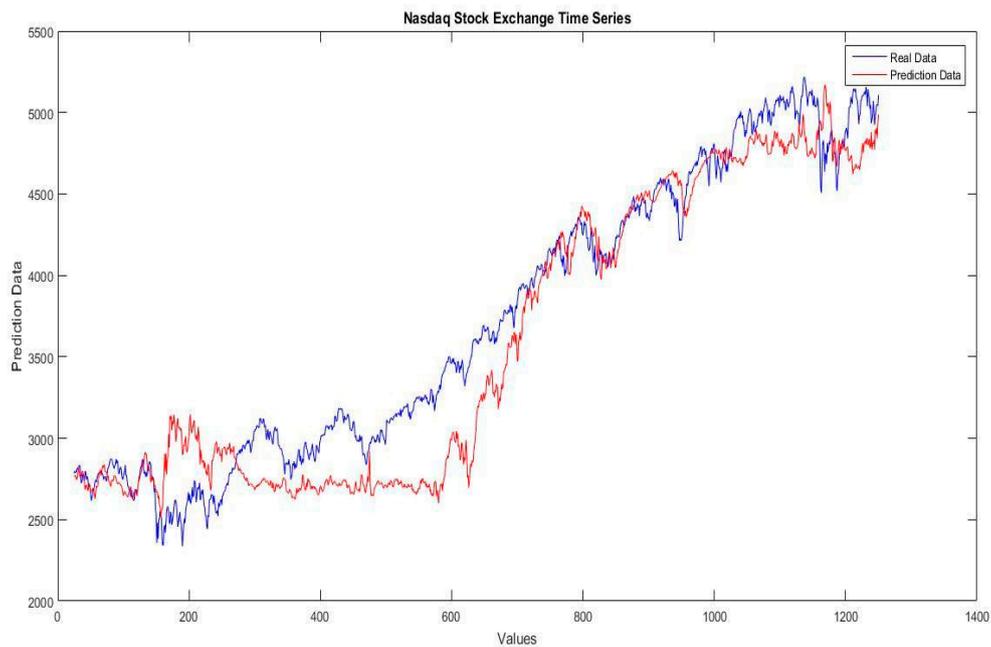


Figure 14. Illustration of the real data against the prediction data of the Nasdaq stock exchange time series for the fuzzy neural network using Hamacher optimized with dynamic PSO.

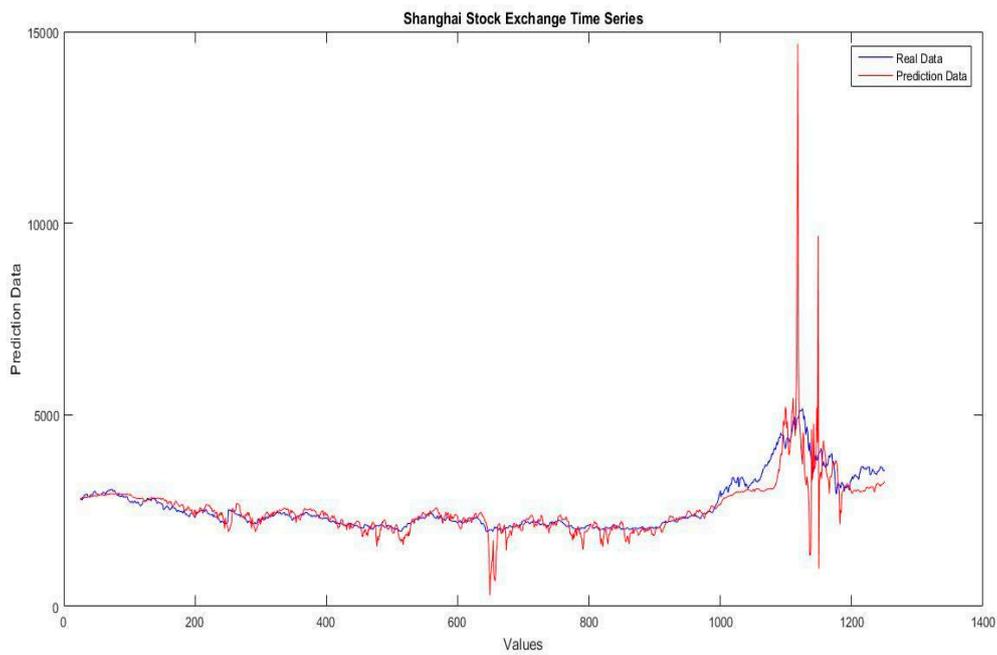


Figure 15. Illustration of the real data against the prediction data of the Shanghai stock exchange time series for the fuzzy neural network using Hamacher and optimized with the dynamic PSO.

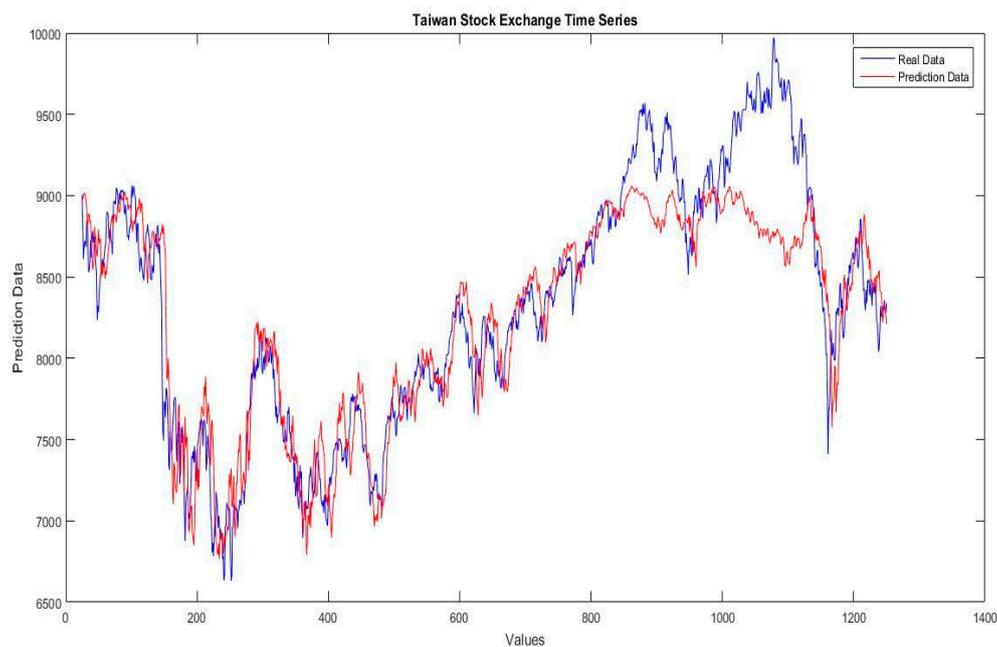


Figure 16. Illustration of the real data against the prediction data of Taiwan stock exchange time series for the fuzzy neural network using Hamacher optimized with dynamic PSO.

A statistical test of T-student is applied to verify the accuracy of the results. In Table 11, the best results for each financial time series is compared with the TNN. The null hypothesis is $H_1 = H_2$ and the alternative hypothesis is $H_1 > H_2$ (AVG = average, SD = standard deviation, SEM = standard error of the mean, ED = estimation of the difference, LLD 95% = lower limit 95% of the difference, GL = degrees of liberty) [69].

Table 11. Comparison of the best results of financial time series prediction with all neural networks.

Financial Time Series	Comparison Results	N	AVG	SD	SEM	ED	LLD 95%	T Value	P Value	GL
Germany	H1:TNN	30	1332.3	22.8	4.2	159.7	66	2.9	0.004	29
	H2: IT2FNWNNH-PSO	30	1173	301	55					
Mexican	H1:TNN	30	935.6	12.5	2.3	−28.15	−35.27	−5.83	1	44
	H2: IT2FNWNNF-PSO	30	9637	23.3	4.3					
Dow-Jones	H1:TNN	30	1082.8	47.1	8.6	−208	−491	−1.24	0.888	29
	H2: IT2FNWNNH-PSO	30	1290	912	167					
London	H1:TNN	30	203.29	2.48	0.45	4.304	2.927	5.24	0.0004	50
	H2: IT2FNWNNH	30	198.98	3.76	0.69					
Nasdaq	H1:TNN	30	558.95	4.19	0.77	28.6	−26	0.89	0.190	29
	H2: IT2FNWNNH-PSO	30	530	176	32					
Shanghai	H1:TNN	30	580	7.48	1.4	36.61	22.93	4.54	0.0004	30
	H2: IT2FNWNNH-PSO	30	543.4	43.5	7.9					
Taiwan	H1:TNN	30	480.11	7.31	1.3	−1.51	−5.88	−0.58	0.717	47
	H2: IT2FNWNNH-PSO	30	481.6	12.2	2.2					

6. Discussion of Results

For the best results of the financial time series, the results in Table 10 show that the interval type-2 fuzzy numbers weights neural network with S-Norms and T-norms Hamacher (IT2FNWNNH-PSO) and Frank (IT2FNWNNF-PSO) optimized using particle swarm optimization (PSO) with dynamic adjustment obtains better results than the neural networks without optimization; only for the London time series is the best result the result of the interval type-2 fuzzy numbers weights neural network with Hamacher (IT2FNWNNH).

The data of the statistical test of T-student in Table 11 demonstrate that the IT2FNWNNH-PSO and IT2FNWNNF-PSO have acceptable accuracy for almost all the financial time series against the traditional neural network. Only in the Nasdaq and Shanghai time series, the T value is minimal to consider a meaningful difference

7. Conclusions

Based on the experiments, we have reached the conclusion that the interval type-2 fuzzy numbers weights neural network with S-Norms and T-norms Hamacher (IT2FNWNNH) and Frank (IT2FNWNNF) optimized using particle swarm optimization (PSO) with dynamic adjustment achieves better results than the traditional neural network and the interval type-2 fuzzy numbers weights neural network without optimization, in almost all the cases of study, for the stock exchanges time series used in this work. This assertion is based on the comparison of prediction errors for all stock exchange time series shown in Table 10 and the data of the statistical test of T-student for best results against the traditional neural network (TNN).

The proposed approach of dynamic adjustment in PSO for optimization of interval type-2 fuzzy numbers weights neural network was compared with the traditional neural network to demonstrate the effectiveness of the proposed method.

The proposed optimization applying an interval type-2 fuzzy inference system to perform dynamic adjustment of the c_1 and c_2 parameters in PSO allows the algorithm find optimal results for the testing problems. In almost all the cases of testing, the results obtained with the IT2FNWNN optimized were the better results, only in the London stock exchanges time series did the IT2FNWNNH obtained better results.

These results are good considering that the number of iterations and the number of particles for the executions of the PSO are relatively small, only 50 iterations with 50 particles. We believe that performing experiments with more iterations and particles will allow it to find better optimal results.

Author Contributions: F.G. worked on the conceptualization and proposal of the methodology, P.M. on the formal analysis, writing and review of the paper, F.V. on the investigation and validation of results, J.R.C. on the software and validation, and A.M.-M. on methodology and visualization.

Funding: This research was funded by CONACYT grant number 178539.

Acknowledgments: The authors would like to thank CONACYT and Tijuana Institute of Technology for the support during this research work.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Wang, X.; Han, M. Online sequential extreme learning machine with kernels for nonstationary time series prediction. *Neurocomputing* **2014**, *145*, 90–97. [[CrossRef](#)]
2. Xue, J.; Zhou, S.H.; Liu, Q.; Liu, X.; Yin, J. Financial time series prediction using $\ell_2,1$ RF-ELM. *Neurocomputing* **2018**, *227*, 176–186. [[CrossRef](#)]
3. Zhou, T.; Gao, S.; Wang, J.; Chu, C.; Todo, Y.; Tang, Z. Financial time series prediction using a dendritic neuron model. *Knowl.-Based Syst.* **2016**, *105*, 214–224. [[CrossRef](#)]
4. Hrasko, R.; Pacheco, A.G.C.; Krohlinga, R.A. Time Series Prediction using Restricted Boltzmann Machines and Backpropagation. *Procedia Comput. Sci.* **2015**, *55*, 990–999. [[CrossRef](#)]
5. Baykasoglu, A.; Özbakır, L.; Tapkan, P. Artificial bee colony algorithm and its application to generalized assignment problem. In *Swarm Intelligence: Focus on Ant and Particle Swarm Optimization*; Felix, T.S.C., Manoj, K.T., Eds.; Itech Education and Publishing: Vienna, Austria, 2007; pp. 113–143.
6. Qin, A.K.; Huang, V.L.; Suganthan, P.N. Differential evolution algorithm with strategy adaptation for global numerical optimization. *IEEE Trans. Evol. Comput.* **2009**, *13*, 398–417. [[CrossRef](#)]
7. Vesterstrom, J.; Thomsen, R. A comparative study of differential evolution, particle swarm optimization, and evolutionary algorithms on numerical benchmark problems. In Proceedings of the CEC2004 IEEE Congress on Evolutionary Computation 2004, Portland, OR, USA, 19–23 June 2004; Volume 2, pp. 1980–1987.
8. Gaxiola, F.; Melin, P.; Valdez, F.; Castillo, O.; Castro, J.R. Comparison of T-Norms and S-Norms for Interval Type-2 Fuzzy Numbers in Weight Adjustment for Neural Networks. *Information* **2017**, *8*, 114. [[CrossRef](#)]
9. Kim, K.; Han, I. Genetic algorithms approach to feature discretization in artificial neural networks for the prediction of stock price index. *Expert Syst. Appl.* **2000**, *19*, 125–132. [[CrossRef](#)]
10. Dash, R.; Dash, P.K. Prediction of Financial Time Series Data using Hybrid Evolutionary Legendre Neural Network: Evolutionary LENN. *Int. J. Appl. Evol. Comput.* **2016**, *7*, 16–32. [[CrossRef](#)]
11. Kim, K. Financial time series forecasting using support vector machines. *Neurocomputing* **2003**, *55*, 307–319. [[CrossRef](#)]
12. Olivas, F.; Valdez, F.; Castillo, O.; Melin, P. Dynamic parameter adaptation in particle swarm optimization using interval type-2 fuzzy logic. *Soft Comput.* **2014**, *20*, 1057–1070. [[CrossRef](#)]
13. Abdelbar, A.; Abdelshahid, S.; Wunsch, D. Fuzzy PSO: A generalization of particle swarm optimization. In Proceedings of the 2005 IEEE International Joint Conference on Neural Networks, Montreal, QC, Canada, 31 July–4 August 2005; Volume 2, pp. 1086–1091.
14. Valdez, F.; Melin, P.; Castillo, O. An improved evolutionary method with fuzzy logic for combining particle swarm optimization and genetic algorithms. *Appl. Soft Comput.* **2011**, *11*, 2625–2632. [[CrossRef](#)]
15. Wu, D.; Tan, W.W. Interval type-2 fuzzy PI controllers why they are more robust. In Proceedings of the 2010 IEEE International Conference on Granular Computing, San Jose, CA, USA, 14–16 August 2010; pp. 802–807.
16. Sepulveda, R.; Castillo, O.; Melin, P.; Rodriguez-Diaz, A.; Montiel, O. Experimental study of intelligent controllers under uncertainty using type-1 and type-2 fuzzy logic. *Inf. Sci.* **2007**, *177*, 2023–2048. [[CrossRef](#)]
17. Gaxiola, F.; Melin, P.; Valdez, F.; Castillo, O. Interval Type-2 Fuzzy Weight Adjustment for Backpropagation Neural Networks with Application in Time Series Prediction. *Inf. Sci.* **2014**, *260*, 1–14. [[CrossRef](#)]
18. Gaxiola, F.; Melin, P.; Valdez, F.; Castillo, O. Generalized Type-2 Fuzzy Weight Adjustment for Backpropagation Neural Networks in Time Series Prediction. *Inf. Sci.* **2015**, *325*, 159–174. [[CrossRef](#)]
19. Durnyak, J.P.; Wunsch, D. Fuzzy regression by fuzzy number neural networks. *Fuzzy Sets Syst.* **2000**, *112*, 371–380. [[CrossRef](#)]

20. Ding, S.; Li, H.; Su, C.; Yu, J.; Jin, F. Evolutionary Artificial Neural Networks: A Review. *Artif. Intell. Rev.* **2013**, *39*, 251–260. [[CrossRef](#)]
21. Weber, S. A general concept of fuzzy connectives, negations and implications based on t-norms and t-conorms. *Fuzzy Sets Syst.* **1983**, *11*, 115–134. [[CrossRef](#)]
22. Hamacher, H. Überlogischeverknüpfungenunscharferaussagen und derenzugehörigebewertungsfunktionen. In *Progress in Cybernetics and Systems Research, III.*; Trappl, R., Klir, G.J., Ricciardi, L., Eds.; Hemisphere: New York, NY, USA, 1975; pp. 276–288.
23. Melin, P.; Olivas, F.; Castillo, O.; Valdez, F.; Soria, J.; Valdez, M. Optimal design of fuzzy classification systems using PSO with dynamic parameter adaptation through fuzzy logic. *Expert Syst. Appl.* **2013**, *40*, 3196–3206. [[CrossRef](#)]
24. Olivas, F.; Valdez, F.; Melin, P.; Sombra, A.; Castillo, O. Interval type-2 fuzzy logic for dynamic parameter adaptation in a modified gravitational search algorithm. *Inf. Sci.* **2019**, *476*, 159–175. [[CrossRef](#)]
25. Kennedy, J.; Eberhart, R. *Swarm Intelligence*; Morgan Kaufmann: San Francisco, CA, USA, 2001.
26. Jang, J.; Sun, C.; Mizutani, E. *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*; Prentice-Hall: Upper Saddle River, NJ, USA, 1997.
27. Engelbrecht, A. *Fundamentals of Computational Swarm Intelligence*; University of Pretoria: Pretoria, South Africa, 2005.
28. Bingül, Z.; Karahan, O. A fuzzy logic controller tuned with PSO for 2 DOF robot trajectory control. *Expert Syst. Appl.* **2001**, *38*, 1017–1031. [[CrossRef](#)]
29. Zadeh, L. The concept of a linguistic variable and its application to approximate reasoning. *Inf. Sci.* **1975**, *8*, 199–249. [[CrossRef](#)]
30. Mendel, J.; John, R. Type-2 fuzzy sets made simple. *IEEE Trans. Fuzzy Syst.* **2002**, *10*, 117–127. [[CrossRef](#)]
31. Liang, Q.; Mendel, J. Interval type-2 fuzzy logic systems: Theory and design. *IEEE Trans. Fuzzy Syst.* **2000**, *8*, 535–550. [[CrossRef](#)]
32. Chandra Shill, P.; Faijul Amin, M.; Akhand, M.A.H.; Murase, K. Optimization of interval type-2 fuzzy logic controller using quantum genetic algorithms. In Proceedings of the IEEE World Congress Computational Intelligence, Brisbane, Australia, 10–15 June 2012; pp. 1027–1034.
33. Karnik, N.N.; Mendel, J.M. Centroid of a type-2 fuzzy set. *Inf. Sci.* **2001**, *132*, 195–220. [[CrossRef](#)]
34. Biglarbegian, M.; Melek, W.W.; Mendel, J.M. On the robustness of Type-1 and Interval Type-2 fuzzy logic systems in modeling. *Inf. Sci.* **2011**, *181*, 1325–1347. [[CrossRef](#)]
35. Mendel, J.M. *Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions*; Prentice Hall PTR: Upper Saddle River, NJ, USA, 2001.
36. Wu, D.; Mendel, J.M. On the continuity of type-1 and interval type-2 fuzzy logic systems. *IEEE Trans. Fuzzy Syst.* **2011**, *19*, 179–192.
37. Ishibuchi, H.; Nii, M. Numerical Analysis of the Learning of Fuzzified Neural Networks from Fuzzy If-Then Rules. *Fuzzy Sets Syst.* **1998**, *120*, 281–307. [[CrossRef](#)]
38. Feuring, T. Learning in Fuzzy Neural Networks. In Proceedings of the IEEE International Conference on Neural Networks, Washington, DC, USA, 3–6 June 1996; Volume 2, pp. 1061–1066.
39. Valdez, F.; Melin, P.; Castillo, O. A survey on nature-inspired optimization algorithms with fuzzy logic for dynamic parameter adaptation. *Expert Syst. Appl.* **2014**, *41*, 6459–6466. [[CrossRef](#)]
40. Li, C.; Wu, T. Adaptive fuzzy approach to function approximation with PSO and RLSE. *Expert Syst. Appl.* **2011**, *38*, 13266–13273. [[CrossRef](#)]
41. Muthukaruppan, S.; Er, M.J. A hybrid particle swarm optimization based fuzzy expert system for the diagnosis of coronary artery disease. *Expert Syst. Appl.* **2012**, *39*, 11657–11665. [[CrossRef](#)]
42. Taher, N.; Ehsan, A.; Masoud, J. A new hybrid evolutionary algorithm based on new fuzzy adaptive PSO and NM algorithms for distribution feeder reconfiguration. *Energy Convers. Manag.* **2012**, *54*, 7–16.
43. Wang, B.; Liang, G.; Wang, C.; Dong, Y. A new kind of fuzzy particle swarm optimization fuzzy PSO algorithm. In Proceedings of the 1st International Symposium on Systems and Control in Aerospace and Astronautics, ISSCAA, Harbin, China, 19–21 January 2006; pp. 309–311.

44. Hongbo, L.; Abraham, M. Fuzzy Adaptive Turbulent Particle Swarm Optimization. In Proceedings of the IEEE Fifth International Conference on Hybrid Intelligent Systems (HIS'05), Rio de Janeiro, Brazil, 6–9 November 2005; pp. 445–450.
45. Shi, Y.; Eberhart, R. Fuzzy Adaptive Particle Swarm Optimization. In Proceedings of the IEEE International Conference on Evolutionary Computation, Seoul, Korea, 27–30 May 2001; IEEE Service Center: Piscataway, NJ, USA, 2001; pp. 101–106.
46. Yang, D.; Li, Z.; Liu, Y.; Zhang, H.; Wu, W. A Modified Learning Algorithm for Interval Perceptrons with Interval Weights. *Neural Process Lett.* **2015**, *42*, 381–396. [[CrossRef](#)]
47. Kuo, R.J.; Chen, J.A. A Decision Support System for Order Selection in Electronic Commerce based on Fuzzy Neural Network Supported by Real-Coded Genetic Algorithm. *Experts Syst. Appl.* **2004**, *26*, 141–154. [[CrossRef](#)]
48. Karnik, N.N.; Mendel, J. Operations on type-2 fuzzy sets. *Fuzzy Sets Syst.* **2001**, *122*, 327–348. [[CrossRef](#)]
49. Chai, Y.; Xhang, D. A Representation of Fuzzy Numbers. *Fuzzy Sets Syst.* **2016**, *295*, 1–18. [[CrossRef](#)]
50. Chu, T.C.; Tsao, T.C. Ranking Fuzzy Numbers with an Area between the Centroid Point and Original Point. *Comput. Math. Appl.* **2002**, *43*, 111–117. [[CrossRef](#)]
51. Dunyak, J.; Wunsch, D. Fuzzy Number Neural Networks. *Fuzzy Sets Syst.* **1999**, *108*, 49–58. [[CrossRef](#)]
52. Coroianu, L.; Stefanini, L. General Approximation of Fuzzy Numbers by F-Transform. *Fuzzy Sets Syst.* **2016**, *288*, 46–74. [[CrossRef](#)]
53. Li, Z.; Kecman, V.; Ichikawa, A. Fuzzified Neural Network based on fuzzy number operations. *Fuzzy Sets Syst.* **2002**, *130*, 291–304. [[CrossRef](#)]
54. Fard, S.; Zainuddin, Z. Interval Type-2 Fuzzy Neural Networks Version of the Stone–Weierstrass Theorem. *Neurocomputing* **2011**, *74*, 2336–2343. [[CrossRef](#)]
55. Molinari, F. A New Criterion of Choice between Generalized Triangular Fuzzy Numbers. *Fuzzy Sets Syst.* **2016**, *296*, 51–69. [[CrossRef](#)]
56. Asady, B. Trapezoidal Approximation of a Fuzzy Number Preserving the Expected Interval and Including the Core. *Am. J. Oper. Res.* **2013**, *3*, 299–306. [[CrossRef](#)]
57. Figueroa-García, J.C.; Chalco-Cano, Y.; Roman-Flores, H. Distance Measures for Interval Type-2 Fuzzy Numbers. *Discret. Appl. Math.* **2015**, *197*, 93–102. [[CrossRef](#)]
58. Requena, I.; Blanco, A.; Delgado, M.; Verdegay, J. A Decision Personal Index of Fuzzy Numbers based on Neural Networks. *Fuzzy Sets Syst.* **1995**, *73*, 185–199. [[CrossRef](#)]
59. Valdez, F.; Vazquez, J.C.; Gaxiola, F. Fuzzy Dynamic Parameter Adaptation in ACO and PSO for Designing Fuzzy Controllers: The Cases of Water Level and Temperature Control. *Adv. Fuzzy Syst.* **2018**, *2018*, 1274969. [[CrossRef](#)]
60. Fan, G.F.; Peng, L.L.; Hong, W.C. Short term load forecasting based on phase space reconstruction algorithm and bi-square kernel regression model. *Appl. Energy* **2018**, *224*, 13–33. [[CrossRef](#)]
61. Dong, Y.; Zhang, Z.; Hong, W.-C. A hybrid seasonal mechanism with a chaotic cuckoo search algorithm with a support vector regression model for electric load forecasting. *Energies* **2018**, *11*, 1009. [[CrossRef](#)]
62. Tung, S.W.; Quek, C.; Guan, C. eT2FIS: An Evolving Type-2 Neural Fuzzy Inference System. *Inf. Sci.* **2013**, *220*, 124–148. [[CrossRef](#)]
63. Castillo, O.; Melin, P. A review on the design and optimization of interval type-2 fuzzy controllers. *Appl. Soft Comput.* **2012**, *12*, 1267–1278. [[CrossRef](#)]
64. Zarandi, M.H.F.; Torshizi, A.D.; Turksen, I.B.; Rezaee, B. A new indirect approach to the type-2 fuzzy systems modeling and design. *Inf. Sci.* **2013**, *232*, 346–365. [[CrossRef](#)]
65. Nguyen, D.; Widrow, B. Improving the Learning Speed of 2-Layer Neural Networks by choosing Initial Values of the Adaptive Weights. In Proceedings of the International Joint Conference on Neural Networks, San Diego, CA, USA, 17–21 June 1990; Volume 3, pp. 21–26.
66. Caraveo, C.; Valdez, F.; Castillo, O. Optimization of fuzzy controller design using a new bee colony algorithm with fuzzy dynamic parameter adaptation. *Appl. Soft Comput.* **2016**, *43*, 131–142. [[CrossRef](#)]
67. Castillo, O.; Neyoy, H.; Soria, J.; Melin, P.; Valdez, F. A new approach for dynamic fuzzy logic parameter tuning in Ant Colony Optimization and its application in fuzzy control of a mobile robot. *Appl. Soft Comput.* **2015**, *28*, 150–159. [[CrossRef](#)]

68. Alexandridis, A.; Sarimveis, H.; Bafas, G. A new algorithm for online structure and parameter adaptation of RBF networks. *Neural Netw.* **2003**, *16*, 1003–1017. [[CrossRef](#)]
69. Derrac, J.; García, S.; Molina, D.; Herrera, F. A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. *Swarm Evol. Comput.* **2011**, *1*, 3–18. [[CrossRef](#)]



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).